APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/904,644	07/13/2001	William H. Roetzheim	COS-01	3382

7590	05/18/2005

William J. Kolegraff
3119 Turnberry Way
Jamul, CA 91935

EXAMINER
VU, TUAN A

ART UNIT	PAPER NUMBER
2193	

DATE MAILED: 05/18/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

	Application No.	Applicant(s)
Office Action Summary	09/904,644	ROETZHEIM, WILLIAM H.
	Examiner	Art Unit	
	Tuan A. Vu	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on *10 January 2005*.

2a) ☒ This action is **FINAL**.   2b) ☐ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) *1-24* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) ☐ Claim(s) _____ is/are allowed.

6) ☒ Claim(s) *1-21 and 23* is/are rejected.

7) ☒ Claim(s) *22 and 24* is/are objected to.

8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.

10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a) ☐ All   b) ☐ Some * c) ☐ None of:

      1. ☐ Certified copies of the priority documents have been received.

      2. ☐ Certified copies of the priority documents have been received in Application No. _____.

      3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage
application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date _____.

4) ☒ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. *20050509* .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

1.   This action is responsive to the Applicant's response filed 1/10/2005.

As indicated in Applicant's response, claims 1, 22, and 24 have been amended. Claims

1-24 are pending in the office action.

Claim Objections

2.   Claims 22 and 24 are objected to because of the following informalities: there appears to

be some symbols being set forth without a explanatory legend in the representation of parametric

variables recited in the parametric rule, e.g. *Env(l), M(a), Life, Std, M(b), KSLOC, Env(s)* (last

line of claim); and the claims should provide a explicit definition of the symbols representing

those variables. Appropriate correction is required.

Allowable Subject Matter

3.   Claims 22 and 24 contain allowable subject matter but are objected to for containing

improprieties pertinent to a proper claim language format as set forth above, but would be

allowable pending correction of such deficiencies.

Some specific exponential elements found in the parametric rule, e.g. $KSLOC^{(Mb + Sum(Envs))}$ – re claim 22; or $Effort ^ {(Tb + Sum(Envs/5))}$ - re claim 24 - are not found to be

obvious from any prior art of record.

Claim Rejections - 35 USC § 103

4.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

5.   Claims 1-12, 15-21, and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable

over Carnegie Mellon Univ., *"Software Engineering Institute Special Report"*: Chap. 6, Chap. 9,

Appendix B, Copyright 1995, CMU/SEI-95-SR-004 (hereinafter Carnegie), in view of

Minkiewicz et al., USPN: 6,073,107 (hereinafter Minkiewicz).

As per claim 1, Carnegie discloses a method of estimating an outcome for a software

development project (SDP), comprising:

selecting a parametric rule having a plurality of variables (ch. 6, Fig. 6-13; *Parametric*

Models, Function Point Model – pg. 34-38; Price Processing, *LH = [...] [...]*, pg. 24, *rules –*

Evidence of credibility – pg. 44; *independent variables* - chp. 9, page 2);

choosing each of a project type (e.g. *systems, WBS* – Chp. 6: Fig. 6.4; Appendix B –

Note: every subsystems define a project type), a lifecycle (e.g. Ch. 6: Fig.6.6, pg. 9; App. B: Fig.

B-1), and a standard (e.g. *MIL-STD-88* - Appendix B: preface; ch.6, pg. 25 -SEER-SEM, *MIL-*

STD-498) for the software development project; using the parametric rule using the variables

being independent (e.g. *independent variables* – para *Parametric Models* - pg. 14) and

generating the outcome (e.g. ch. 6, *Cocomo II outputs* - pg. 22; *Cocomo 81 outputs* – pg.

18).

But Carnegie does not expressly disclose assigning a type factor responsive to choosing

the project type; assigning a lifecycle factor responsive to choosing the lifecycle; or assigning a

standard factor responsive to choosing the standard; nor does Carnegie disclose using the type

factor, the lifecycle factor, and the standard factor as variables in the parametric rule. Carnegie

uses different model approaches to make variables out of considered from project type, stage of

development and generate output using rules and associating variables therefrom (standard:

MIL-STD, ANSI J-STD, type: COTS... database, lifecycle: *development method ... waterfall*,

see ch. 6, SEER-SEM pg. 25-34); hence has suggested using variables from project type,

standard and lifecycle, i.e. a quantized factor representing those variables derived from type,

standard, or lifecycle stage. The use of variables or factor representing a real world requirement

entities (e.g. domain, environment, complexity, application type – Cocomo 81, Cocomo II) in a

parametric rule for assessing and evaluating possible outcome of a software project was a

known-concept at the time the invention was made (e.g. COCOMO 81, COCOMO II).

Carnegie further discloses independent factor like time, size, etc. to further relate to dependent

variables to cost estimation equation (*independent variables* - chp. 9, page 2). Enhancing

Carnegie's approach to provide cost variables and noncost factors like size, rate, time, project

standard as above mentioned, Minkiewicz, in a method to estimate cost using parameters

analogous to the techniques (e.g. COCOMO, ch. 6 pg. 14-23) mentioned by Carnegie, discloses

visual wizard using variables (e.g. lifecycle activity, application) represented by parameters

being collected inside a wizard / tool used to calculate forecast project cost or variation (e.g.

Type: *Application Type*, lifecycle: *Lifecycle Model: Medium*, standard: *ISO 9000*, Fig. 6-12).

Thus, it would have been obvious for one of ordinary skill in the art at the time the invention was

made to take the variables as application/project type, lifecycle stage, and software project

standard as inputs and associate a factor thereto, then provide these factors as parameters being

collected for implementing a parametric rule factoring those parameters as suggested by

Minkiewicz because, like other factors such as those suggested by COCOMO in the generation

of parameters for SDP cost output, these variables would also be represented in some quantized

values or numerical factors such as exemplified by Minkiewicz, such to enable the estimation be

formulated in a mathematical/parametric rule yielding a quantized outcome or numerical output.

Nor does Carnegie does not explicitly disclose the type factor, the lifecycle factor, and

the standard factor as variables in the independent parametric rules even though Carnegie

mentions that some parameters like cost is a dependent variable from other variables; but the

usefulness of these factors have been obvious as addressed above. Further, in Carnegie

estimation relationship model (CER chp. 9), it is disclosed that some dependent variables like

cost can be estimated from *cost drivers* or independent variables such as cost elements relating

performance/physical characteristics, project time; or non-cost elements like weight, volume,

lines of code (see **Introduction**: *independent variables* - chp. 9, page 2); and this suggests that

project characteristics like type, a lifecycle representation, or a standard are non-cost driving type

of parameters. Hence, in case Carnegie does not already use type, lifecycle, and standard factors

as these independent cost driver elements, it would have been obvious for one of ordinary skill in

the art at the time the invention was made to provide these factors as independent parametric

elements so to help dependent variables like cost in the CER as intended by Carnegie (combined

with Minkiewicz) to provide estimate of dependent variables – or cost - on basis of independent

variables for a project where lifecycle, type and standard have been a cost driving factors as set

forth above.

As per claim 2, Carnegie teaches effort being an outcome (e.g. *PM* – Cocomo II pg. 18-

19; *Effort* -Fig. 6-18, pg. 39-40).

As per claim 3, see Carnegie: *Defects*, Fig. 6-18, pg. 39-40.

As per claim 4, see Carnegie: *Schedule*, Fig. 6-18, pg. 39-40.

As per claim 5, see Effort, Cost, Fig. 16-8 (Note: the notion of measuring estimated cost as an outcome is inherent to COCOMO and similar tool like Minkiewicz's).

As per claim 6, Carnegie discloses process maturity and required schedule factors (e.g. Fig. 6-10, pg. 17; Fig. 6-11, pg. 20) hence has suggested a lifecycle-related factor according to the lifecycle methodology and capability of a SDP (see section B-E, pg. 3-12) and association of those factors with a table or a checklist (section E, pg. 12). Hence, Carnegie has suggested a look-up table for associating lifecycle related factors in conjunction with generating a lifecycle parameter used in the parametric rule implementation. Minkiewicz, in the parameters collecting wizard, further discloses table representing different lifecycle stages and related numbers (Fig. 11-12). It would have been obvious for one of ordinary skill in the art at the time the invention was made to provide a table as suggested by Carnegie, so that corresponding numerical factors representing a stage of a SDP lifecycle can be looked up as suggested via Minkiewicz's teachings. This would enable a fast retrieval of the corresponding parameters for a specific factor destined to be used for implementing the parametric rule, and in so doing expedite the generation or enable the automation of such rule-based outcome generation, all this in conjunction with the common benefit that goes with using a look-up table as taught by well-known practices.

As per claim 7, the limitation of using a look-up table for a standard factor would have been obvious in view of the rationale in addressing the standard factor in claim 1 combined with that addressing the look-up table limitation of claim 6.

As per claim 8, Carnegie does not explicitly teaches a lifecycle factor being a linear parameter; but indirectly includes factor that emanates from choosing a lifecycle model (see

claim 6); while Minkiewicz teaches parameters related to an lifecycle stage (Fig. 9-12). The

motivation for combining the lifecycle factors as taught by Minkiewicz to the rule-based

approach by Carnegie (to generate estimate cost output – Cocomo 81, Cocomo II, re claim 1)

such that those factors are linear factor of an parametric equation or formula as suggested by

Cocomo and enhanced by Minkiewicz would have been obvious and established for the same

benefits in using the factors representing the lifecycle variable as addressed in claim 1.

As per claim 9, the limitation of using a standard factor as a linear parameter would

have been obvious in view of the rationale in addressing the standard factor in claim 1 combined

with that addressing the linear lifecycle factor limitation of claim 8.

As per claim 10, Carnegie discloses complexity being a factor (Fig. 6-10) and

Minkiewicz discloses parametric values to be inversely proportional with the ascending order of

a lifecycle evolution (Fig. 11-12). Official notice is taken that the time consumed or resources

being spent on the earlier stages of a SDP is more significant than those spent at a latter stages

(e.g. requirement stages versus maintenance stage) of the SDP was a known concept at the time

the invention was made. Hence, it would have been obvious for one of ordinary skill in the art at

the time the invention was made to made lifecycle factor as being inversely proportional to an

outcome, e.g. cost decrease as lifecycle stage ascends because of the known concept learned

from spending requirement engineering resources up front as well-known in the art of SDP.

As per claim 11, whereas the type of project such as a project for military with a specific

standard as noted by Carnegie dictates the cost in proportion of resources (e.g. *DOD, 80:20,*

20:80 - Introduction pg. 1; *MIL-STD*, pg. 25), the standard factor can play some linear role in

setting a parametric rule. Carnegie does not specifically teach providing a standard factor as an

inverse factor in the parametric rule. But given the complexity of a military project being a factor

and the effect of a hardware statistics involved as a military standard is applied as implemented

by Carnegie, it would have been obvious for one of ordinary skill in the art at the time the

invention was made to provide the standard factor, a military factor, in a parametric formula (as

from the combined teachings of Carnegie and Minkiewicz) such that it is inversely proportional

to an outcome, e.g. software cost versus hardware cost as suggested above because of the reasons

as mentioned from military complex combination of hardware involvement as opposed to a

predominantly software cost in a non-military SDP.

As per claim 12, Carnegie discloses a size being a number of lines of code (e.g. *KSLOC*

– Cocomo II pg. 19, 21; *SLOC* - Figure 6-18, pg. 39).

As per claim 15, Carnegie (combined with Minkiewicz) discloses an environment factor

(e.g. SEER-SEM inputs: *Platform, environment complexity*, Target – pg. 25-26).

As per claim 16, Carnegie teaches a WBS (Appendix B) and suggests Case tools for

collecting user's specified requirements (e.g. *Case Tool*, D. Object Points – pg. 33) and

Minkiewicz discloses a wizard with a interface viewer tool including Case tool(e.g. Fig. 7-13) to

collect metrics for the parametric evaluation. Although Carnegie and Minkiewicz do not

explicitly teach a template for generating a product breakdown, the teachings from above

suggests using a template like interface for setting up the breakdown of parameters prior to

establishing the relationships between factors in order to produce the estimation outcome from

the parametric rule implementation. Hence, it would have been obvious for one of ordinary skill

in the art at the time the invention was made to modify the breakdown as suggested by Carnegie

using a generic template tool as taught by Minkiewicz or suggested by Carnegie's Case Tool, so

that a factor, e.g. lifecycle factor, can be displayed and user-manipulated via the template; and

the motivation for this would have been because such template would allow user interaction and

editing capability in regard to data being entered for specifying an user's requirements, using one

of the most commonly known features of today's graphical development tool such as Case

Tools as taught above.

As per claim 18, the rationale for rejection of the limitation of a generic standard

template is the same as that set forth in claim 16 above.

As per claims 17 and 19, these claims relate to a form of intended use of the generic

template created for the lifecycle and the standard as set forth in claim 16 and 18 above, hence

the use of a template for allowing user interaction, dynamic modification and selection of items

on such template is but one obvious intended use thereof. Thus, the claims would have been

obvious for the same reasons as set forth in claims 16, 18, by virtue of the implied use of

template for mapping a selection onto items created for that very purpose.

As per claim 20, this claim includes a type factor, a lifecycle factor, a standard factor as

recited in claim 1 in conjunction with a parametric rule; and further includes the environmental

factor and size element as addressed in claims 15 and 12, respectively. The limitation for using

all those factors into the parametric rule would have been obvious in light of the rationale used to

address said factors the corresponding claims above.

As per claim 21, Carnegie only teaches effort in terms of factors as a laid out by Cocomo

approach (re claim 1) but in view of the rationale as to use all the factors listed in claim 20, the

general form of EFFORT = FACTOR * LIFECYCLE * STANDARD * ENVIRONMENT *

SIZE would have been also obvious in view of the rationale of claims 1, 15, and 12.

As per claim 23, Carnegie does not disclose 'Defect = defect factor * effort * (1/lifecycle factor) * (1/standard factor)'. However, Carnegie teaches Military standard and CMM levels (Fig. 6-7, pg. 10) and computation of defects (e.g. PRICES S: ...*delivered defects* – pg. 25; SEER-SEM Processing: ... *defects are computed* – pg. 27); while Minkiewicz discloses the decreasing of cost as the lifecycle stages advances and defect statistics (Defect/Func – Fig. 11, 12). Official notice is taken that the concept that the more advanced the maturity level is or the more solid a development standard is defined, the less chance of cost incurred due to software errors or verification failure would be noted was particularly well-known in the art of developing software project at the time the invention was made. Hence, it is therefore evident that the maturity levels in conjunction with Carnegie's military standards as mentioned above would play a factor such that they fall under the rationale of the well-known concepts as noticed above. In combination with the inverse linearity relationship taught by Minkiewicz in terms of the lifecycle, it is recognized that the estimate for defect would decrease when lifecycle factor increases, and when the standard factor is more advanced. Following of the rationale used in claim 10 and combining it with the above defect estimate dependency on lifecycle and standard, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the defect computation as taught by Carnegie and the linear factor arrangement as suggested by COCOMO's parametric-based calculating of outcome so to include the inverse relationship of lifecycle and standard as noted above in conjunction with the effort for providing the defect calculation thus mentioned. The motivation would be because this parametric rule would exhibit how much the defect factor is influenced by the complexity demanding a linear

increase in effort as opposed to it being dependent on the inverse relation with lifecycle and

standard elements, as analyzed from above notice.

6.   Claims 13 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Carnegie Mellon Univ., *"Software Engineering Institute Special Report"*: Chap. 6, Appendix B,

Copyright 1995, and Minkiewicz et al., USPN: 6,073,107, as applied to claim 12, further in view

of Srulowitz et al., "Software Estimation", URL:

www.saspin.org/SASPIN_Apr2001_COCOMO.pdf (hereinafter Minkiewicz).

As per claims 13 and 14, Carnegie only teaches SLOC being generated with Function

points techniques (e.g. Cocomo II, SEER-SEM – pg. 19-33). Using also function points to

establish SLOC output, Srulowitz, in a method for estimation of software using the approach

based on COCOMO like Carnegie, discloses using internet points and Domino points (Metric

Methodologies Supported – pg. 43). The evolution of network transmission and internet-based

medium for carrying business application and industrial, military project data such that collecting

data being distributed across those media and estimating size thereof by means of internet-

adapted techniques (*internet points* – re claim 13, *Domino points* – re claim 14) for estimation

was a known concept at the time the invention was made. Hence, it would have been obvious for

one of ordinary skill in the art at the time the invention was made to provide such techniques as

interpoint points or Domino points techniques shown by Srulowitz to compute the SLOC as

suggested by Carnegie because this would enable the distributed application data under the SDP

to be capture according to a techniques differentiated to operate with the client-server or browser

application of a given distributed SDP, as taught by known concepts and suggested by Srulowitz.

Response to Arguments

7.   Applicant's arguments filed 1/10/2005 have been fully considered but they are not

persuasive. Following are the Examiner's response thereto.

(A)   Applicants have submitted that Carnegie does not teach 'choosing each of a project type,

a lifecycle, and a standard" (Appl. Rmrks, pg. 8, bottom; pg. 9, top) and that the cited portions by

Examiner do not show that the project type, a lifecycle, and a standard are chosen for a

parametric rule (Appl. Rmrks, pg. 10, bottom para). The rejection has shown a cost estimate

model type of software project using models and using equation in a parametric relationship and

rules to govern how these parametric components are used in the process of estimating; hence

has fulfilled the parametric rule limitation.   The rejection has also shown parametric rules in that

the estimation cost model use independent variables as well as dependent variables (see

pertinent parts of ch. 6 and ch. 9). The claim recites selecting a rule which is to interpreted as a

model being implemented with different variables being defined therein as dependent or

independent variables as shown by Carnegie; such model being implemented via assigning these

variables in a parametric rule or equation. The rejection has shown all these limitations except

for the very nature of these variables for which a combination with the teaching by Minkiewicz

is needed. Based on the rationale that independent variables are supporting the cost variables

which are dependent on other parameter/variables, the rejection has now shown how combining

factors viewed as independent factors like type, lifecycle, or standard as mentioned by Carnegie

in the cost estimate model (parametric rule using equation) would have been obvious. The

Applicants have to show why such combination would have been inappropriate and for what

specific reasons so. Applicants argue that choosing maturity levels by Carnegie does not teach

these above factors being chosen for a parametric rule. The claim is unclear as to what

parametric rule amounts to and as long as a model shows some cost estimating factors

relationship established via for examples of equation in which parameters or variables are used,

the parametric rule and the selecting are disclosed. Further, the concept of *rule* is broad and can

be equated to evaluation/estimating model using arithmetical approaches or constructs as shown

by Carnegie's or COCOMO's estimation cost equation in the cited portions.

(B) Applicants have submitted that none of the examples by COCOMO by Carnegie teaches

choosing a standard, type, and lifecycle (Appl. Rmrks, pg. 11, middle para) and that Carnegie's

knowledge base dictates assigning of parameters (Appl. Rmrks, pg. 12, top para) as opposed to

choosing each of the type, lifecycle and standard as recited. The claim is not providing sufficient

teaching as to why choosing parameters or type variables in a parametric rule would be different

from applying some equation wherein some parameters are independent or dependent as in

Carnegie's model for cost estimation. The rejection has combined suggestion of using type or

standard by Carnegie as well as independent variables teaching in the cost evaluation rule set for

enabling the constructs in COCOMO's parametric rule to be combined with teachings by

Minkiewicz to enable variables as application/project type, lifecycle stage, and software project

standard as inputs to the COCOMO's rule in which suggestion about standard or type has been

put forth. The arguments fail to show why such combined teachings would have been adverse

and teaching away from their own settings or purposes. In other words, Applicants fail to show

why it would not be obvious for one skill in the art to take Minkiewicz's size, effort rate or even

factors like standard as independent parameters to further enhance the equation as taught by

Cocomo, in view of Carnegie's use of model setting forth cost factors and cost driving

independent variables. Besides, Applicants seem to dissect one reference in a piecemeal manner

-- in light of Applicants' perception of the claimed invention, not from the viewpoint of possible albeit different interpretations by one skill in the art -- when the claim language is lacking specificity and particularly when the rejection is based on a USC 103 obviousness type of combination of teachings. In response to applicant's arguments against the references taken individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

(C)   Applicants have submitted that Minkiewica's input parameters is not same as choosing of project type, lifecycle and a standard (Appl. Rmrks, pg. 12, middle para) and that neither Carnegie or/and Minkiewicz disclose independent variables like type, lifecycle or standard. There is no specificity in the claim as to enforce that *independent variables* is to be interpreted in only one way, i.e. independent variables means only one very specific thing, notwithstanding the fact that the term *independent* bears some degree of relativity in itself (*independent* with respect to what). The rejection has shown what qualifies as independent and what as dependent. And the arguments have to address the rejection in light of a combination as set forth therein and pinpointing the places where particular teachings put together are inappropriate; not as individual reference taken individually. In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

Conclusion

8.   **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time

policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within TWO

MONTHS of the mailing date of this final action and the advisory action is not mailed until after

the end of the THREE-MONTH shortened statutory period, then the shortened statutory period

will expire on the date the advisory action is mailed, and any extension fee pursuant to 37

CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event,

however, will the statutory period for reply expire later than SIX MONTHS from the mailing

date of this final action.

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Tuan A Vu whose telephone number is (272) 272-3735. The

examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Kakali Chaki can be reached on (571)272-3719.

The fax phone number for the organization where this application or proceeding is

assigned is (571) 273-3735 (for non-official correspondence – please consult Examiner before

using) or 703-872-9306 (for official correspondence) or redirected to customer service at 571-

272-3609.

Any inquiry of a general nature or relating to the status of this application should be

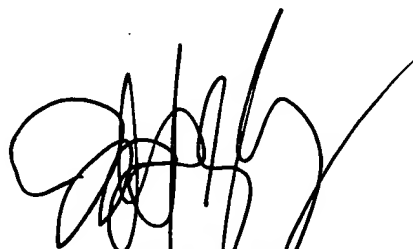directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

VAT
May 10, 2005

TODD INGBERG
PRIMARY EXAMINER